International Journal of Computer Science Engineering
and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831; ISSN(E): 2249-7943
Vol. 3, Issue 5, Dec 2013, 95-102


# NAME ENTITY RECOGNITION ON PUNJABI LANGUAGE

**KULJOT SINGH**

Bhai Gurdas Institute of Engineering & Technology, Sangrur, Punjab, India

## ABSTRACT

This paper is about Name Entity Recognition (NER) for Punjabi Language. Lot of work has been done on English language but not much on Indian languages, particularly on Punjabi.ConditionalRandom field approach has been used for developing NER system.We are presenting the result 85.78% F-Score of our experiment by adding some useful features in ConditionalRandom field such as Three Word Window and Bigram on a baseline of 80.92%[1].

**KEYWORDS:** NER, CRF on Punjabi, Punjabi Language, Bigram

## INTRODUCTION

Named entity recognition involves locating and classifying the names in text which are known as Name Entities [5][6]. NER is an important task, having applications in information extraction (IE), question answering (QA), machine translation and in most other NLP applications. NER involves the identification of named entities such as person names, location names, names of organizations, monetary expressions, dates, numerical expressions etc. A variety of techniques has been used for NER. The three major approaches to NER are:

- Linguistic approaches.
- Machine learning (ML) based approaches.
- Hybrid approach.

The linguistic approaches typically use rules manuallywritten by linguists. There are several rule based NER systems, containing mainly lexicalized grammar, gazetteer lists, and list of trigger words, which are capable of providing 88%-92% f-measure accuracy for English 8][15][19]. The main disadvantages of these rule-based techniques are that these require huge experience andgrammatical knowledge of the particular languageor domain and these systems are not transferable to other languages or domains.

ML based techniques for NER make use of alarge amount of NE annotated training data to acquirehigh level language knowledge. Several ML techniques have been successfully used for the NER task of which hidden markov model [3], maximum entropy [4], conditional random field [14][2][17] are widely used.

Hybrid technique is the combination of both the Linguistic and Machine learning approach. This approach has been successfully implemented by various authors. It has been used on Indian languageswhich was designed for theInternational Joint Conference on Natural Language Processing(IJCNLP) and Named Entity Recognition for South and South East Asian Languages (NERSSEAL) shared task, that applies maximum entropy model, language specific rules and gazetteers to the task of named entity recognition (NER) and 65.13% f-value in Hindi, 65.96% f-value in Bengali and44.65%, 18.74%, and 35.47%f-value in Oriya, Telugu and Urdu respectively was obtained[10]. NER systems use gazetteer lists for identifying names. Both thelinguistic approach [19][8]and the ML based approachause gazetteer lists[4][18].

NER system is deeply explored by some of the great authors, NERSSEAL has played an important role in developing NER system among different languages. ConditionalRandom Field (CRF) was experimented on differentIndian languages like Bengali, Hindi, Urdu, telugu[4]and for telugu separately[17],Hybrid techniques was also implemented in that area [11][16]. The NER task for Hindi has been explored which used morphological and contextual evidences [7], the system achieved 41.70% f-value with a very low recall of 27.84% and about 85% precision. A more successful Hindi NER system was developed with feature induction [12]. They were able to achieve 71.50% f value using a training set of size 340k words. More results were found in Hindi [9]. Their maximum entropy markov model (MEMM) based model gives 79.7% f-value. A great F-score has also been calculated on Punjabi, which is the highest and first ever result on Punjabi language [1].

## CRF MODEL

Conditional random fields (CRFs) are a class of statistical modeling method often applied in pattern recognition and machine learning, where they are used for structured prediction. Whereas an ordinaryclassifier predicts a label for a single sample without regard to "neighboring" samples, a CRF can take context into account; e.g., the linear chain CRF popular in natural language processing predicts sequences of labels for sequences of input samples.

CRFs are a type of discriminative undirected probabilisticgraphical model. It is used to encode known relationship between observations and construct consistent interpretations. It is often used for labeling orparsing of sequential data, such as natural language text or biological sequences and in computer vision.

CRF model is a simple, customizable, and open source implementation of Conditional Random Fields (CRFs) for segmenting/labeling sequential data. CRF++ (tool kit) is designed for generic purpose and will be applied to a variety of NLP tasks, such as Named Entity Recognition, Information Extraction and Text Chunking.

Conditional Random Fields (CRFs) are undirected graphical models, a special case of which corresponds to conditionally-trained finite state machines. CRFs are used for labeling sequential data. In the special case in which the output nodes of the graphical model are linked by edges in a linear chain, CRFs make a first-order Markov independence assumption, and thus can be understood as conditionally-trained finite state machines (FSMs). Let *o = (o, o2, o3, o4, oT )* be some observed input data sequence, such as a sequence of words in text in a document,(the values on n input nodes of the graphical model). Let S be a set of FSM states, each of which is associated with a label, l? £.Let *s = (s1, s2, s3 ,s4 ,. sT)* be some sequence of states, (the values on T output nodes). By the Hammersley- Clifford theorem, CRFs define the conditional probability of a state sequence given an input sequence to be:

$$P(s|o) = \frac{1}{Z_o} * exp(\sum_{t=1}^{T}\sum_{k}\lambda_k f_k(s_{t-1}, s_t, o, t))$$

Where *Zo* is a normalization factor over all statesequences is an arbitrary feature function over it sarguments, and? *k* is a learned weight for each feature function. A feature function may, for example, be defined to have value 0 or 1. Higher? Weights make their corresponding FSM transitions morelikely. CRFs define the conditional probability of a label sequence based on the total probability overthe state sequences,

$$P(l|o) = \sum_{s:l(s)=l} P(s|o)$$

Where l*(s)* is the sequence of labels corresponding to the labels of the states in sequence ***s***.

$$Z_o = \sum_{s \in S^T} \exp\left(\sum_{t=1}^{T} \sum_{k} \lambda_k f_k(s_{t-1}, s_t, \mathbf{o}, t)\right),$$

Note that the normalization factor, Zo, (also known in statistical physics as the partition function) is the sum of the scores of all possible states.

And that the number of state sequences is exponential in the input sequence length T. In arbitrarilystructured CRF's calculating the normalization factor in closed form is intractable, but in liner chain- structure CRFs, the probability that a particulartransition was taken between two CRF states at a particular position in the input can be calculated by dynamic programming.

**Unigram and Bigram Features**

These are two important features of template file used in performing NER on the input data i.e. Training and Test Files [10] with CRF++.

Unigram template: first character, **'U'**. This is a template to describe unigram features. When you give a template "U01:%x[0,1]", CRF++ automatically generates a set of feature functions (func1... funcN). The number of feature functions generated by a template amounts to (L * N), where L is the number of output classes and N is the number of unique string expanded from the given template.

Bigram template: first character, **'B'**. This is a template to describe bigram features. With this template, a combination of the current output token and previous output token (bigram) is automatically generated. Note that this type of template generates a total of (L * L * N) distinct features, where L is the number of output classes and N is the number of unique features generated by the templates. When the number of classes is large, this type of templates would produce at on of distinct features that would cause inefficiency both in training/testing.

## NAME ENTITY RECOGNITION IN PUNJABI

Punjabi is an Indo-Aryan language spoken by 130 million (2013 estimate) native speakers worldwide, making it the 9th most widely spoken language (2010) in the world. In India it is spoken normally in Punjab state.

Partially NE tagged Punjabi news corpus developed from the archive of a widely read daily ajit Punjabi news paper[1].The corpus contains around 19 lacks word forms in UTF-8 format. A portion of thispartially NE tagged corpus has been manually annotatedwith the four NE tags [11].

## A NAMED ENTITY TAGSET

The training data of Punjabi language is annotated with Four NE tagswhich has been represented in Conference on Computational Natural Language Learning (Co NLL 2003) shared task i.e. person name, location name, organization name and miscellaneous[14].

## TRAINING DATA

Preparation of training data has been done with some preprocessing of each word annotating with their respective tags. The annotated data uses *IOB* [10] formatted text in which a *B-XXX* tag indicates the first word of an entity type *XXX* and *I-XXX* is used for subsequent words of an entity. The tag *O* indicates the word is outside of a NE.

The training data for Punjabi contains more than 17k words in which four feature tags have been defined [11].

Earlier also some work has been done on Punjabi language by using conditional random fields approach [1].

Our problem is to improve the result of existing approach on Punjabi language by adding some useful features of it.

## NAMED ENTITY FEATURES

Feature selection plays a crucial role in CRF framework. Experiments were carried out to find out most suitable features for NE tagging task. The main features for the NER task have been identified based on the different possible combination of available word and tag context. The features also include prefix and suffix for all words. The term prefix/suffix is a sequence of first/last few characters of a word, which may not be a linguistically meaningful prefix/suffix. The use of prefix/suffix information works well for highly inflected languages like the Indian languages. In addition, various gazetteer lists have been developed to use in the NER task particularly for Punjabi. We have considered different combination for the NER task:

Following is the details of the set of features that were applied to the NER task:

### Context Word Feature

Previous and next words of a particular word might be used as a feature. Wehave considered the word window of size three, i.e., previous and next word from the current word

### Word Suffix

Word suffix information is helpful to identify NEs. A fixed length word suffix of the current and surrounding words might be treated as feature. In this work, suffixes of length up to three the current word have been considered for all the languages. More helpful approach is to modify the feature as binary feature. Variable length suffixes of a word can be matched with predefined lists of useful suffixes for different classes of NEs.

### Word Prefix

Prefix information of a word is alsohelpful. A fixed length prefix of the current and the surrounding words might be treated as features. Here, the prefixes of length up four have been considered for all the language.

### Gazetteer Lists

Various gazetteer lists have beencreated from a tagged punjabi news corpus for Punjabi [1]. The first, last and middle names of person has been taken from the daily Ajit news website. The person name collections had to be processed in order to use it in the CRF framework. The simplestapproach of using these gazetteers is to comparethe current word with the lists and make decisions.

### Parts of Speech (POS) Information

We have also used the Parts of Speech (POS) of the current and/or the surrounding word(s) as features. Here we have used a rule-based POS tagger [13] developed by Punjabi University. This tagger uses fine-grained tagset with around 630 tags. For our evaluation, we have used a highly coarse-grained tagset with the following tags which are NN(Noun), PN(Pronoun), AJ(Adjective), AV(Adverb), Preposition(PP), Conjunction(CJ), Interjection(IJ) and PT(Postposition). Although POS tagger is very helpful in tagging the data but the success of the task is limited by the accuracy of this tagger. The wrong tags were manually corrected for NER task.

### Named Entity Information

The NE tag of the previous word is also considered as a feature. This is the only dynamic feature in the experiment.

## EXPERIMENTAL SETUP

### Training File Preparation

Two important categories i.e. Training file and Test file have been built in order to perform NER through Condition based approach.

### Evaluation Matrices

The results are presented in the form ofrecall(R), precision (P) and F-measure percentage.They are defined as follows:

$$\text{Recall} = \frac{\text{correct entities recognized}}{\text{Total correct entities}}$$

$$\text{Precision} = \frac{\text{correct entities recognized}}{\text{Total entities recognized}}$$

$$\text{F-measure} = \frac{2 \text{ x recall x precision}}{\text{Recall + precision}}$$

### Feature Sets

Feature set indicates the set of name entity features. We have selected the best feature set in which the highest f-score has been achieved.Following table represent the feature set taken for the process along with comparison of baseline result.

**Table 1**

<table>
<tr><th>Feature Set</th><th>F-Score Value with Three Word Window and Bigram</th><th>Without Bigram and Third Word Window</th></tr>
<tr><td>pw,cw,nw,Bigram</td><td>59.50</td><td>56.52</td></tr>
<tr><td>pw,cw,nw,pt,Bigram</td><td>77.70</td><td>71.72</td></tr>
<tr><td>pw,cw,nw,pp,cp,np,pt,Bigram</td><td>86.01</td><td>76.62</td></tr>
<tr><td>pw,cw,nw,pp,cp,np,Bigram</td><td>69.63</td><td>62.97</td></tr>
<tr><td>pw,cw,nw,pt,pp,cp,np,0<|prefix|<4, 0<|suffix|<4,</td><td>86.14</td><td>80.05</td></tr>
<tr><td>pw,cw,nw,pt,pp,cp,np,1<|prefix|<5,1<|suffix|<5, Person-Prefix List, Bigram</td><td>86.05</td><td>79.84</td></tr>
<tr><td>pw,cw,nw,pt,pp,cp,np,1<|prefix|<5,1<|suffix|<5, First Name List, Bigram</td><td>85.93</td><td>80.69</td></tr>
<tr><td>pw,cw,nw,pt,pp,cp,np,1<|prefix|<5,1<|suffix|<5, First name ,Middle name, Last Name List, Bigram</td><td>85.78</td><td>80.92</td></tr>
<tr><td>pw,cw,nw,pt,cp,np,1<|prefix|<5,1<|suffix|<5,First name List ,Middle name , Last Name,Person-Prefix List,Day& Month List, Bigram</td><td>85.56</td><td>80.90</td></tr>
<tr><td>pw,cw,nw,pt,cp,np,1<|prefix|<5,1<|suffix|<5,First name List ,Middle name , Last Name,LocationList,Person-Prefix List,Day& Month List, Bigram</td><td>85.63</td><td>80.37</td></tr>
</table>

Notations used for the feature setsare:

**cw, pw, nw**: Current, previous and next word.

**cwi, pwi, nwi:**Current, Previous and the next ithord from the current word.

**prefix|,|suffix|:** Length of Prefix and suffix of the current word.

**pt:** NE tag of the previous word.

**cp, pp, np:** POS tag of the current, previous and the next word.

**cpi, ppi, npi:** POS tag of the current, previous and the next ith word from the current word.

## CONCLUSIONS AND FUTURE SCOPES

We have prepared a CRF based system for theNER task on Punjabilanguage. We have also addedsome useful features in CRF. Also our derived rules need to be modified for improvement of the system. As the sizeof training data is not much for this language, rules and gazetteers would be effective. We have experimented with CRF model only, other ML methods like HMM, MaxEnt or MEMM may be able to give better accuracy.

## REFERENCES


1. AmandeepKaur, Gurpreet Singh Josan and JagroopKaur. 2009 Named Entity Recognition for Punjabi: A Conditional Random Field Approach. *In Proceedings of 7th international conference on Natural Language ProcessingICON-09.*Macmillan Publishers, India.
2. AsifEkbal, et.al. Language Independent Named Entity Recognition in Indian Languages. *In Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages,* pages 33–40, Hyderabad, India, January 2008. c 2008 Asian Federation of Natural Language Processing.
3. Bikel Daniel M., Miller Scott, Schwartz Richard and Weische del Ralph. 1997. Nymble: A High Performance Learning Name-finder. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, 194–201.
4. Borthwick Andrew. 1999. A Maximum Entropy Approach to Named Entity Recognition. *Ph.D. thesis, Computer Science Department, New York University*.
5. Chinchor, Nancy. 1995. MUC-6 Named Entity Task Definition (Version 2.1). *MUC-6*, Columbia, Maryland.
6. Chinchor, Nancy. 1998. MUC-7 Named Entity Task Definition (Version 3.5). *MUC-7*. Fairfax, Virginia.
7. CucerzanSilviu and Yarowsky David. 1999. Language Independent Named Entity Recognition Combining Morphological and Contextual Evidence. In *Proceedings of the Joint SIGDAT Conference on EMNLP and VLC 1999*, 90–99.
8. Grishman Ralph. 1995. The New York University System MUC-6 or Where's the syntax? In *Proceedings of the Sixth Message Understanding Conference*.
9. KarthikGali, et. al. Aggregating Machine Learning and Rule Based Heuristics for Named Entity Recognition *Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages, pages 25–32,* Hyderabad, India, January 2008. c 2008 Asian Federation of Natural Language Processing.

10. Kumar Saha, Chatterji 2008. A Hybrid Approach for Named Entity Recognition in Indian Languages. *In Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages*, pages 17–24, Hyderabad, India, January 2008.

11. Kumar N. and Bhattacharyya Pushpak. 2006. Named Entity Recognition in Hindi using MEMM. In *Technical Report, IIT Bombay, India.*.

12. Li Wei and McCallum Andrew. 2004. Rapid Development of Hindi Named Entity Recognition using Conditional Random Fields and Feature Induction (Short Paper). In *ACM Transactions on Computational Logic*.

13. Mandeep Sing Gill, Gurpreet Singh Lehal and Shiv Sharma Joshi. Parts-of-Speech Tagging for Grammar Checking of Punjab, *In the Linguistics Journal Volume 4 Issue 1, pages 6-22.*

14. McCallum, A.; W. Li. 2003. Early Results for Named Entity Recognition with Conditional Random Fields, Feature Induction and Web-Enhanced Lexicons. In *Proceedings CoNLL-03*, Edmanton, Canada.

15. McDonald D. 1996. Internal and external evidence in the identification and semantic categorization of proper names.In *B. Boguraev and J. Pustejovsky, editors, Corpus Processing for Lexical Acquisition*, 21–39.

16. Praveen Kumar P Ravi Kiran V A Hybrid Named Entity Recognition System for South Asian Languages. *In Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages*, pages 83–88, Hyderabad, India, January 2008. c 2008 Asian Federation of Natural Language Processing.

17. Praneeth M Shishtla, KarthikGali, Prasad Pingali and VasudevaVarma Experiments in Telugu NER: A Conditional Random Field Approach.*In Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages,* pages 105–110, Hyderabad, India, January 2008. c 2008 Asian Federation of Natural Language Processing.

18. Srihari R., Niu C. and Li W. 2000.A Hybrid Approach for Named Entity and Sub-Type Tagging. In *Proceedings of the sixth conference on Applied natural language processing*.

19. Wakao T., Gaizauskas R. and Wilks Y. 1996. Evaluation of an algorithm for the recognition and classification of proper names.In *Proceedings of COLING-96*.